Table des matières

[Documentation du Projet : Analyse des Performances des Équipes de Hockey 2](#_Toc189248252)

[I. Lien GitHub : https://github.com/latifnjimoluh/tpchill.git 2](#_Toc189248253)

[II. Documentation Technique 2](#_Toc189248254)

[1.1 Objectifs du Projet 2](#_Toc189248255)

[1.2 Technologies Utilisées 2](#_Toc189248256)

[1.3 Structure des Fichiers 3](#_Toc189248257)

[1.4 Installation et Lancement 3](#_Toc189248258)

[1.5 Fonctionnalités Techniques 3](#_Toc189248259)

[III. Documentation Fonctionnelle 3](#_Toc189248260)

[1.1 Objectifs et Fonctionnalités 3](#_Toc189248261)

[2.1 Utilisation de l'Application 4](#_Toc189248262)

[3.1 Réponses aux Questions du Projet 5](#_Toc189248263)

# Documentation du Projet : Analyse des Performances des Équipes de Hockey

# Lien GitHub : <https://github.com/latifnjimoluh/tpchill.git>

## Documentation Technique

### Objectifs du Projet

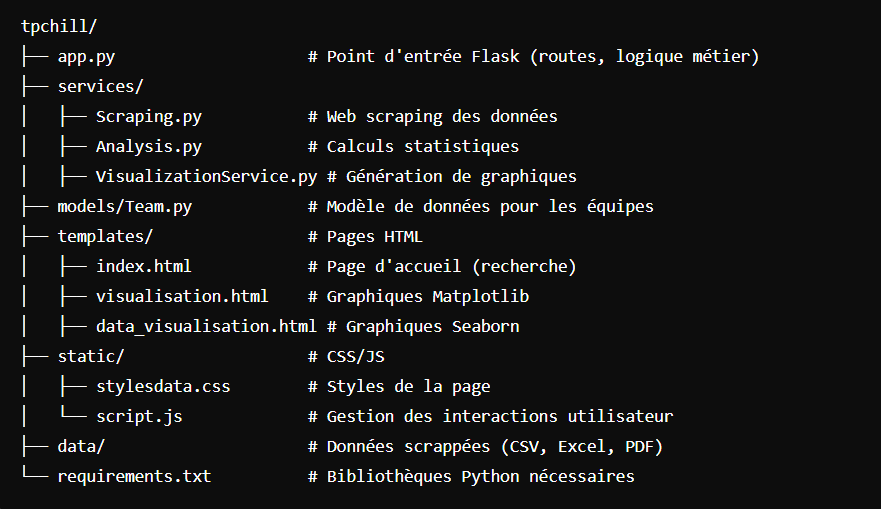
Le projet vise à développer une application web interactive pour analyser les performances historiques des équipes de hockey. Les principales tâches comprennent :

* **Web Scraping** pour extraire les données des équipes depuis Scrape This Site.
* **Analyse statistique** des performances avec Pandas et NumPy.
* **Visualisation interactive** des données avec Matplotlib et Seaborn.
* **Interface web** construite avec Flask pour permettre une interaction fluide.

### Technologies Utilisées

| **Catégorie** | **Outils** |
| --- | --- |
| **Web Scraping** | BeautifulSoup, requests |
| **Analyse des données** | Pandas, NumPy |
| **Visualisation** | Matplotlib, Seaborn |
| **Backend** | Flask |
| **Frontend** | HTML/CSS/JavaScript |
| **Gestion de fichiers** | CSV, Excel, PDF (via reportlab) |

### Structure des Fichiers



### Installation et Lancement

#### Cloner le dépôt :

git clone **https://github.com/latifnjimoluh/tpchill.git**

cd **tpchill**

#### Installer les dépendances :

pip install -r **requirements.txt**

#### Lancer l’application :

python app.py

#### Accéder à l’application :

* + Ouvrir <http://localhost:8080> dans un navigateur.

### Fonctionnalités Techniques

* **Web Scraping** : Les données sont extraites des pages du site "Scrape This Site" à l'aide de BeautifulSoup et requests. Les informations récupérées sont sauvegardées sous forme de fichiers CSV.
* **Analyse Statistique** : Utilisation de Pandas et NumPy pour calculer des statistiques descriptives (moyenne, médiane, etc.) et des corrélations (Pearson) entre les différentes variables (victoires, buts marqués, etc.).
* **Visualisation** : Des graphiques sont générés à l'aide de Matplotlib et Seaborn, incluant des boxplots, des scatter plots et des heatmaps.
* **Backend (Flask)** : L'application est construite avec Flask pour gérer les routes, afficher les résultats et servir les graphiques générés.
* **Frontend (HTML/CSS/JavaScript)** : L’interface web permet à l’utilisateur de rechercher des équipes, afficher les résultats et télécharger les données sous différents formats (CSV, Excel, PDF).

## Documentation Fonctionnelle

### Objectifs et Fonctionnalités

Le projet permet d'analyser les performances des équipes de hockey à travers une interface web interactive. Les utilisateurs peuvent rechercher des équipes spécifiques et consulter des analyses statistiques sur leurs performances. Les fonctionnalités comprennent :

1. Recherche d'Équipes (Page : /):
   * Permet de rechercher une équipe en entrant son nom dans un champ de saisie.
   * Affichage dynamique des statistiques de l’équipe : victoires, défaites, buts marqués, etc.
   * Les résultats sont paginés pour faciliter la navigation.
   * Téléchargement des résultats au format CSV, Excel ou PDF.
2. Analyse Statistique (Page : /analyse):
   * Calcul des statistiques descriptives (moyenne, médiane, écart-type).
   * Calcul de la corrélation entre les victoires et les buts marqués (Pearson).
   * Identification de la meilleure année de performance d’une équipe.
3. Visualisation des Données avec Matplotlib (Page : /visualisation):
   * **Graphiques Matplotlib** : Évolution des victoires d’une équipe, histogrammes des buts marqués.
4. Visualisation des Données avec Seaborn (Page : /data\_visualisation):
   * **Graphiques Seaborn** : Boxplot des victoires, heatmap des performances par année, scatter plot des buts marqués et pourcentage de victoires.
5. Comparaison des Équipes (Page : /data\_visualisation):
   * Sélection de plusieurs équipes et d’une période pour comparer leurs performances sur différents critères (victoires, buts marqués, etc.).
   * Génération de graphiques comparatifs (lignes, barres) pour visualiser les performances.
6. Téléchargement des Données (Disponible sur toutes les pages de résultats) :
   * Les résultats peuvent être téléchargés en formats CSV, Excel ou PDF via les options disponibles dans l’application.

### Utilisation de l'Application

1. Rechercher une équipe (Page : /) :
   * Entrez un nom d’équipe (ex : "Canadiens de Montréal") dans le champ de recherche et cliquez sur "Rechercher".
   * Les résultats de la recherche seront affichés avec des statistiques paginées.
2. Analyser les performances (Page : /analyse) :
   * Allez sur la page /analyse pour consulter les statistiques descriptives et les corrélations des performances de l’équipe.
3. Comparer les équipes (Page : /data\_visualisation) :
   * Accédez à la page /data\_visualisation pour sélectionner plusieurs équipes et une période, puis visualisez les graphiques comparatifs.
4. Visualiser les données avec Matplotlib (Page : /visualisation) :
   * Consultez cette page pour visualiser des graphiques comme l’évolution des victoires et les histogrammes des buts marqués.
5. Visualiser les données avec Seaborn (Page : /data\_visualisation) :
   * Visualisez des boxplots, des heatmaps et des scatter plots qui aident à analyser les performances des équipes sur différentes périodes.
6. Télécharger les résultats (Disponible sur toutes les pages avec résultats) :
   * Cliquez sur le bouton "Télécharger" pour exporter les résultats en CSV, Excel ou PDF.

### Réponses aux Questions du Projet

1. Corrélation victoires/buts marqués : Calculée à l'aide de la méthode pearsonr dans Analysis.py. Cette information est affichée dans l’onglet Analyse (/analyse).
2. Meilleur ratio de victoires : Calculé avec la fonction best\_teams\_by\_win\_ratio() dans Analysis.py. Cette fonction affiche les 5 meilleures équipes avec leur pourcentage moyen de victoires, consultable via la page /analyse.
3. Comparaison des performances : Dans /data\_visualisation, l’utilisateur peut sélectionner plusieurs équipes et une période pour comparer leurs performances via des graphiques générés dynamiquement avec DataVisualizationService.py.